# AUTOMATIC LOOP-SHAPING OF QFT CONTROLLERS VIA LINEAR PROGRAMMING[1]

Yossi Chait[2] Qian Chen[3] and C.V. Hollot[4]

## ABSTRACT

In this paper we focus on the following loop-shaping problem: Given a nominal plant and QFT bounds, synthesize a controller that achieves closed-loop stability, satisfies the QFT bounds and has minimum high-frequency gain. The usual approach to this problem involves loop shaping in the frequency domain by manipulating the poles and zeroes of the nominal loop transfer function. This process now aided by recently-developed computer-aided design tools, proceeds by trial and error, and its success often depends heavily on the experience of the loop-shaper. Thus, for the novice and first-time QFT users, there is a genuine need for an automatic loop-shaping tool to generate a first-cut solution. Clearly, such an automatic process must involve some sort of optimization, and, while recent results on convex optimization have found fruitful application in other areas of control design, their immediate usage here is precluded by the inherent non-convexity of the QFT bounds. Alternatively, these QFT bounds can be over-bounded by convex sets, as done in some of the recent approaches to automatic loop-shaping, but this conservatism can have a strong and adverse effect on meeting the original design specifications. With this in mind, we approach the automatic loop-shaping problem by first stating conditions under which QFT bounds can be dealt with in a non-conservative fashion using linear inequalities. We will argue that for a first-cut design, these conditions are often satisfied in the most critical frequencies of loop-shaping and are violated in frequency bands where approximation leads to negligible conservatism in the control design. These results immediately lead to an automated loop-shaping algorithm involving only linear programming techniques, which we illustrate via an example.

---

[1] Supported by a National Science Foundation grant No. CMS-9313764. This support is appreciated.

[2] Associate Professor, Mechanical & Industrial Engineering Department, University of Massachusetts, Amherst, MA 01003, USA; chait@ecs.umass.edu.

[3] Ph.D. research assistant, Mechanical & Industrial Engineering Department, University of Massachusetts, Amherst, MA 01003, USA; qchen@ecs.umass.edu.

[4] Associate Professor, Electrical & Computer Engineering Department, University of Massachusetts, Amherst, MA 01003, USA; hollot@ecs.umass.edu.

## 1. INTRODUCTION

The Quantitative Feedback Theory (QFT) method offers a direct, frequency-domain based design approach for tackling feedback control problems with robust performance objectives. In this approach, the plant dynamics may be described by frequency response data, or by a transfer function with mixed (parametric and non-parametric) uncertainty models. One feature that distinguishes QFT from other frequency-domain methods, such as $H_\infty$ and LQG/LTR, is its ability to deal directly with uncertainty models and robust performance criteria. This is achieved by translating robust performance specifications and uncertainty models into so-called QFT bounds. These bounds, typically displayed on a Nichols chart-like plot, then serve as a guide for shaping the nominal loop transfer function which involves the manipulation of gain, poles and zeros. This design process is executed efficiently using computer-aided design software, such as the QFT Control Design MATLAB Toolbox (Borghesani *et al.*, 1995), and is effective for "simple"[5] problems. Nevertheless, QFT designers are often challenged by such control problems due to a lack of loop-shaping experience, and could benefit from an algorithm that automatically provides a first-cut solution to the loop-shaping problem. In addition, an automatic loop-shaping facility would enhance the capabilities of the expert QFT designer. Automatic loop-shaping algorithms have been proposed over the past twenty years and this paper reports on a new version.

One of the first papers to address the automatic loop-shaping problem is Gera and Horowitz (1980). This work used Bode's gain-phase integral to derive a nominal loop shape in an iterative fashion. There was no guarantee of convergence and rational function approximation was ultimately needed to obtain an analytical expression for the loop. This approach was automated in a QFT Toolbox (Ballance and Gawthrop, 1991) which simplified the iteration process and allowed for higher order approximations of the integral. In Thompson and Nwokah (1994), automatic loop-shaping was achieved using nonlinear programming techniques where the QFT bounds were overbound by disks[6] As with any nonlinear optimization technique, this approach may be sensitive to initial conditions and fail to converge to a global optimum. Also, closed-loop stability was not

[5] The loop-shaping task is challenging when the plant has unstable poles, nonminimum-phase zeros, delays or a large number of resonances, or, when the control problem involves tradeoff between competing specifications.

[6] To generate these disks, the QFT specifications were converted into standard $H_\infty$ weights on the closed-loop sensitivity function. While the resulting QFT bounds are now disks , this conversion clearly results in design conservatism suggesting an $H_\infty$ design.

assured. Avoiding these optimization difficulties, Bryant and Halikias (1995) applied linear programming techniques to the automatic loop-shaping problem. However, this comes at the expense of introducing conservatism in describing non-convex QFT bounds with linear inequalities. In addition this approach cannot deal with unstable poles and zeroes in the loop transfer function nor does it guarantee closed-loop stability. Finally, Zhao and Jayasuriya (1993) introduced the Youla parameterization to transform a QFT robust performance problem into a one-dimensional search; but this allows for only one controller parameter to be automatically designed.

In this paper we provide an automatic loop-shaping algorithm that builds upon the previously described work. We pose the loop-shaping problem as a linear program, which yields a stabilizing controller of prescribed order and minimal hi-frequency gain. In contrast to Bryant and Halikias (1995), the QFT bounds are tightly described by linear inequalities. This is achieved by first posing the QFT problem in terms of the closed-loop complementary sensitivity function T rather than the nominal loop transfer function as done in the classical QFT approach. Then, since these (closed-loop) QFT bounds are not generally convex, we transform the problem so that they can be exactly described by linear inequalities. This transformation constitutes one of the technical novelties of this paper. These convex QFT bounds are then evaluated at a finite set of frequencies to form a set of linear inequalities constraining T. Next, closed-loop stability is imposed by fixing the poles of T (to be stable). Finally, a linear program is solved where the cost measures hi-frequency controller gain and where the linear constraints (non-conservatively) represent the closed-loop QFT bounds. A key limitation of our approach is that the poles of T are fixed with only the zeros taken as optimization variables.

Finally, we note that in the earliest rigorous investigation, Bailey *et al*. (1994) compared the open-loop gain-phase shaping approach (i.e., conventional QFT) with the closed-loop gain-phase shaping approach (i.e., the basis of our new procedure). They noted the difficulty in the latter approach when the corresponding "bounds" are not convex. And with the design conservatism related to convexification of such bounds one may question the utility of the design itself. However, as we propose in this paper, if the purpose of automatic loop-shaping is to present the designer with a reasonable initial loop design, then this approach is merited. The power of QFT lies in its ability to guide the designer in gain-phase tuning of the open-loop frequency response especially around the crucial crossover range. Indeed, our experience shows that manual tuning using powerful computer-aided tools (e.g., Borghesani, *et al.*, 1995) can overcome such conservatism.

This paper is organized as follows. In the next section we outline the classical QFT loop-shaping problem and repose it in terms of the closed-loop complementary sensitivity function T. We also present our main technical result, which suggests a transformation under which the resulting closed-loop QFT bounds can be described by linear inequalities. In Section 3, we bring these elements together and pose the automatic loop-shaping problem as a specific linear program. Finally, in Section 4, we provide an illustrative example to demonstrate the utility of the proposed algorithm.

## 2. CONVEXITY OF CLOSED-LOOP QFT BOUNDS

In this section we briefly outline the classical QFT problem and introduce the technical novelty of this paper concerned with the convexity of closed-loop QFT bounds. The starting point for QFT design is a negative unity feedback system where the plant $P(s)$ is modeled by a family of transfer functions

$$\mathsf{P} = \left\{ P(s) = \frac{k}{(s+a)(s+b)} : \quad k \in [1,10], a \in [1,5], b \in [20,30] \right\}$$

The feedback problem is to design a controller G(s) such that the closed-loop system is robust stable and

$$\left| \frac{PG(j\omega)}{1+PG(j\omega)} \right| \leq \quad = 1.2, \quad \text{for all } P \in \mathsf{P}, \quad \omega \in [0, \infty)$$

Ignoring robust stability here, the above algebraic specification can be translated into an equivalent specification on the frequency response of the nominal open-loop $P_0G(j\omega)$ (for some $P_0(s) \in \mathsf{P}$ ). For example, at $\omega = 100$, if $P_0G(j\omega)$ lies outside the region shown in Figure 1, then the above specification is met ($P_0$ here corresponds to $[k_0,a_0,b_0]$ = [1,1,20]) Such regions are called QFT bounds and are typically shown on a Nichols chart for loop-shaping purposes. Clearly, this bound is not convex, a fact posing a real problem in automatic loop-shaping procedures.

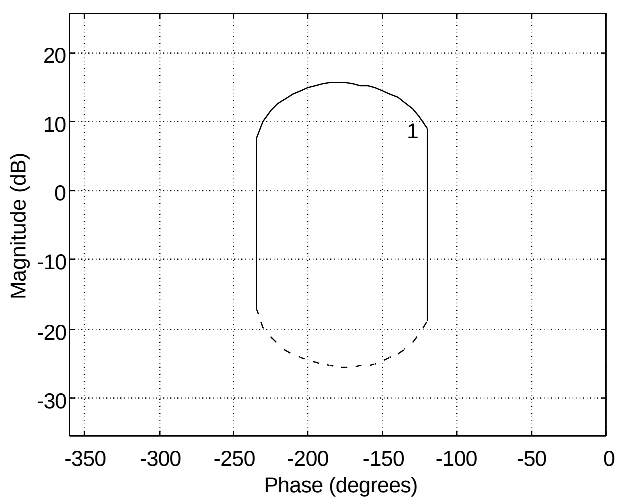


Figure 1: QFT bound $\omega = 100$

At this point we are ready to formally define QFT bounds. Consider the bounds at a single frequency where we can ignore stability aspects and focus on (sets of) complex numbers. Let $\mathcal{P} \subset \mathcal{C}$ be the closed set (i.e., template) where the open-loop plant is allowed to vary in; i.e., $P \in \mathcal{P}$. Let $\mathcal{T}$ denote the closed set that describes the specification on the complementary sensitivity function $T \in \mathcal{T}$. The design problem amounts to finding a controller C that leads to

$$-\frac{PC}{1+PC} \in \mathcal{T} \quad \forall P \in \mathcal{P} \tag{1}$$

Now introduce the bilinear mapping $f : z \in \mathcal{C} \rightarrow w \in \mathcal{C}$

$$w = f(z) = \frac{z}{1+z}$$

and its inverse $g : w \rightarrow z$

$$z = g(w) = \frac{w}{1-w}$$

Then,

$$f(PC) \in \mathcal{T} \quad \Leftrightarrow \quad PC \in g(\mathcal{T})$$

and (1) becomes

$$PC \in g(\mathcal{T}) \qquad \forall P \in \mathcal{P}$$

Let us assume, without loss of generality, that $0 \notin \mathcal{P}$. Then, the above equation is equivalent to

$$C \in \frac{1}{P} g(\mathcal{T}) \quad \forall P \in \mathcal{P}$$

The latter is nothing but

$$C \in \bigcap_{P \in \mathcal{P}} \frac{1}{P} g(\mathcal{T}) \tag{2}$$

Now, specify a nominal plant $P_0$. Then (2) is rewritten as

$$P_0 C \in \bigcap_{P \in \mathcal{P}} \frac{P_0}{P} g(\mathcal{T}) \tag{3}$$

The set on the right-hand-side of (3) is a so-called *QFT bound* $\mathcal{B}_{P_0}$; i.e.,

$$\mathcal{B}_{P_0} \equiv \bigcap_{P \in \mathcal{P}} \frac{P_0}{P} g(\mathcal{T})$$

Generally speaking, QFT bounds are not necessarily convex sets. However, as we show below, by mapping the QFT bounds $\mathcal{B}_{P_0}$ into bounds on the nominal closed-loop complementary sensitivity:

$$T_0 \equiv f(P_0 C) = \frac{P_0 C}{1 + P_0 C} \tag{4}$$

convexity can sometimes be achieved and a sufficient condition for convexity can be derived . Specifically, the set

$$\mathcal{B}_{T_0} \equiv \left\{ f(P_0 C):\ P_0 C \in \mathcal{B}_{P_0} \right\} \tag{5}$$

can be convex even when $\mathcal{B}_{P_0}$ is not. Then, the QFT loop-shaping task is transformed to one on $T_0$, i.e.,

$$P_0 C \in \mathcal{B}_{P_0} \quad \Leftrightarrow \quad T_0 \in \mathcal{B}_{T_0}$$

The following result gives a sufficient condition for $\mathcal{B}_{T_0}$ to be convex.

**Theorem 1**. *Assume $\mathcal{T} \subset \mathbb{C}$ is a closed disk centered at the origin. If there exists a complex number c such that*

$$-1 \notin \frac{c}{P} g(\mathcal{T}) \qquad \forall P \in \mathcal{P} \tag{6}$$

*then* $\mathcal{B}_{T_0}$ *is a convex set by selecting* $P_0=c$.

**Proof**: Because the set $\mathcal{T}$ is a closed disk centered at the origin and $g$ is bilinear, then $\frac{c}{P}g(\mathcal{T})$ is either a disk or the complement of a disk. Moreover, since $-1\notin \frac{c}{P}g(\mathcal{T})$, for all $P \in \mathcal{P}$, then $\frac{c}{P}g(\mathcal{T})$ does not contain the critical point of the bilinear map $f$. Hence, $f\left(\frac{c}{P}g(\mathcal{T})\right)$ is a closed disk for any $P\in\mathcal{P}$.

Now taking $P_0 = c$, we have

$$\mathcal{B}_{T_0} \equiv \bigcap_{P\in\mathcal{P}} f\left(\frac{c}{P}g(\mathcal{T})\right)$$

Finally, since the intersection set of convex sets is itself a convex set, we have shown that $\mathcal{B}_{T_0}$ is convex. •

Next, we derive an alternative result based on Nichols charts and QFT templates, which reduces verification of (6) to a simple matter of graphical observation. Before we present this result, we require some notation and a definition.

Recall that the Nichols chart is a modified polar representation of the complex plane where each complex number $v = x + iy$ has polar representation $v = re^{i\phi}$. The corresponding point on the Nichols chart is $(\phi,\rho)$ where $\rho = 20\,log\, r$. The map

$$n(v):(x,y) \rightarrow (\phi,\rho)$$

transforms the Cartesian plane into the Nichols chart.

**Definition 1:** Let $\mathcal{Q} \subset \mathcal{C}$ be a closed subset of the Nichols chart. Then $\tilde{\mathcal{Q}}$ is the *symmetric set of* $\mathcal{Q}$ if

1) Given $x \in \mathcal{Q}$, there exists a $y \in \tilde{\mathcal{Q}}$ such that $x + y = n(-1,0)$.
2) Given $y \in \tilde{\mathcal{Q}}$, there exists a $x \in \mathcal{Q}$ such that $x + y = n(-1,0)$.

If $\mathcal{Q}$ is a subset of the Nichols chart, let $\overline{\mathcal{Q}}$ denote its set complement. The following theorem gives a sufficient condition for convexity of $\mathcal{B}_{T_0}$ that can be visually verified in the Nichols chart.

**Theorem 2**. *Let $\mathcal{G}$ denote the symmetric set of $\overline{n(\mathrm{g}(\mathcal{T}))}$. If there exists a finite complex number c such that*

$$\mathrm{n(c)\text{-}n}(\mathcal{P}) \subseteq \mathcal{G} \qquad (7)$$

*then $\mathcal{B}_{T_0}$ is a convex set.*

**Proof**: First,

$$n\left(\frac{\mathrm{c}}{\mathrm{P}}\, g(\mathcal{T})\right) = n(\mathrm{c}) - n(\mathrm{P}) + n(g(\mathcal{T})) \qquad \forall\, \mathrm{P} \in \mathcal{P}$$

so that

$$\overline{n\left(\frac{\mathrm{c}}{\mathrm{P}}\, g(\mathcal{T})\right)} = n(\mathrm{c}) - n(\mathrm{P}) + \overline{n(g(\mathcal{T}))} \qquad \forall\, \mathrm{P} \in \mathcal{P} \qquad (8)$$

Now, assume (7) holds. From Definition 1 and (8) it follows that

$$n(-1,0) \in \overline{n\left(\frac{\mathrm{c}}{\mathrm{P}}\, \mathrm{g}(\mathcal{T})\right)} \qquad \forall \mathrm{P} \in \mathcal{P}$$

Thus

$$n(-1,0) \notin n\left(\frac{\mathrm{c}}{\mathrm{P}}\, \mathrm{g}(\mathcal{T})\right) \qquad \forall \mathrm{P} \in \mathcal{P}$$

Consequently,

$$-1 \notin \frac{\mathrm{c}}{\mathrm{P}}\, g(\mathcal{T}) \qquad \forall \mathrm{P} \in \mathcal{P}$$

and using Theorem 1 $\mathcal{B}_{T_0}$ is a convex set. •

Finally, in certain cases we can simplify the verification of condition (7). Indeed, if $\overline{n(g(\mathsf{T}))}$ is symmetric about the Nichols chart line $\{(\phi,\rho):\phi=-180^{o},-\infty<\rho<\infty\}$, then (7) is equivalent to

$$n(\mathsf{P})\text{-}\ n(\mathrm{c})\subseteq\overline{n(g(\mathsf{T}))}\qquad(9)$$

We summarize this result in the following corollary.

**Corollary 1**. *Assume* $\overline{n(g(\mathsf{T}))}$ *is symmetric about the Nichols chart line* $\{(\phi,\rho):\phi=-180^{o},-\infty<\rho<\infty\}$, *then* $\mathsf{B}_{\mathrm{T}_0}$ *is a convex set if (9) holds.*

We illustrate our results using a simple example adapted from Borghesani *et al*., (1995). The plant family is defined by :

$$\left\{\frac{\mathrm{k}}{(\mathrm{s}+\mathrm{a})(\mathrm{s}+\mathrm{b})}:\quad \mathrm{k}\in[1,10],\ \mathrm{a}\in[1,5],\ \mathrm{b}\in[20,30]\right\}$$

In Fig 2, we show the Nichols chart plant templates $n(\mathsf{P})$ at 0.1 and 100 rad/sec. The specification $\mathsf{T}$ is the set $\{\mathrm{T}:\ |\mathrm{T}|\leq 1.05\}$ which is a disk. In the Nichols chart, the boundary of $(g(\mathsf{T}))$ is the classic M-circle. The area inside the M-circle is exactly $\overline{n(g(\mathsf{T}))}$ which is symmetric about the Nichols chart line $\{(\phi,\rho):\phi=-180^{o},-\infty<\rho<\infty\}$ so Corollary 1 can be used to establish the convexity of $\mathsf{B}_{\mathrm{T}_0}$.

To do this, we will determine if condition (9) is satisfied. In Fig 2 we illustrate possible translations of $n(\mathsf{P})$ at 0.1 and 100 rad/sec. Specifically, we show that by proper choice of the parameter c in Theorem 1, $n(\mathsf{P})$ at 100 rad/sec can be translated to inside the M-circle, which is exactly $\overline{n(g(\mathsf{T}))}$. The choice of c = 9.0142e-5+i3.203e-5 corresponding to the nominal plant $\mathrm{P}_0$ with k = 1, b = 30, and a = 5 ($\mathrm{P}_0$ = c), satisfies condition (9) and the bound $\mathsf{B}_{\mathrm{T}_0}$ at 100 rad/sec is indeed convex (see Fig 3).

It is important to understand that this choice of $\mathrm{P}_0$ is nontrivial. In many QFT publications, it is recommended that the nominal plant be selected from the boundary of the template $n(\mathsf{P})$ ((Rodrigues, et. al, 1997). But we follow this reasoning, there may be no such $\mathrm{P}_0$ leading to convex bound. In general, the optimal choice is the plant whose response lies inside the template centered near its lower end. In this example, $n(\mathsf{P})$ easily fits in $\overline{n(g(\mathsf{T}))}$ implying that we have greater flexibility in selecting the parameter c. However, taking another $P_0$ with k = 5, b = 25, a = 3 results in a nonconvex $\mathsf{B}_{T_0}$ (Fig. 4). On the other hand, the $n(\mathsf{P})$ at 0.1 rad/sec can not be translated to inside the M-circle implying that condition (9) cannot be satisfied. Indeed, Fig. 5 shows $\mathsf{B}_{T_0}$ to be nonconvex. However, at the low frequency, the overriding specification is typically the high gain type. High-gain specifications are the focus of the next result.

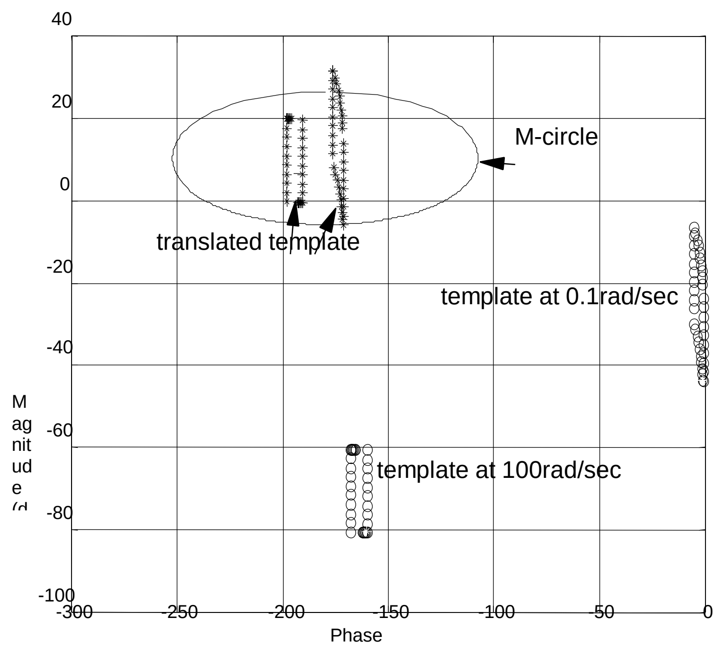


Figure 2: Templates at 0. 1 and 100 rad/sec and $\overline{n(g(\mathsf{T}))}$

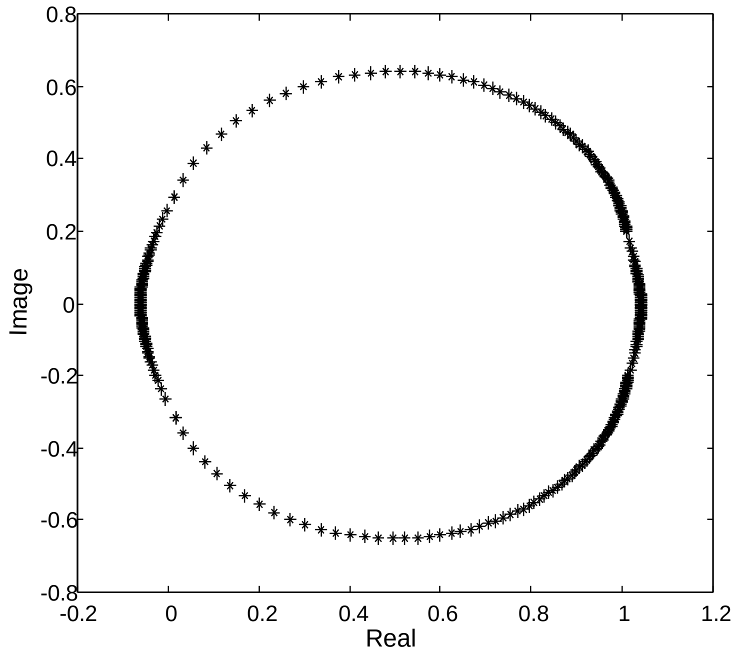


Figure 3: $B_{T_0}$ at 100 rad/sec in the complex plane with $P_0$ = 9.0142e-5+i3.203e-5

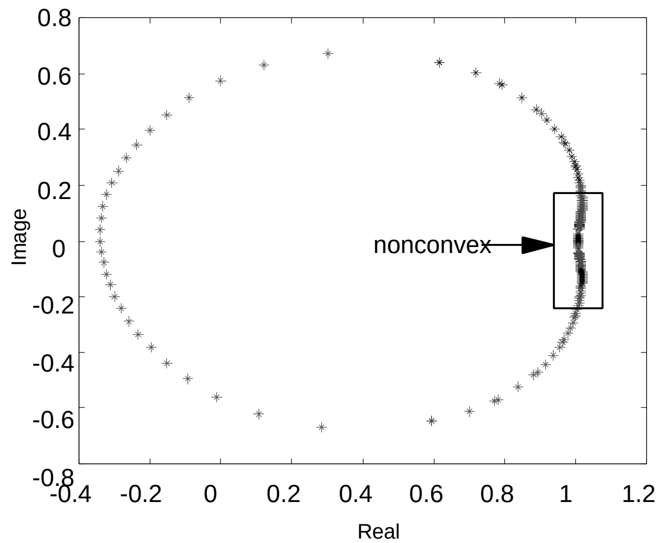


Figure 4: $B_{T_0}$ at 100 rad/sec in the complex plane with $P_0$ = 5+0.1i

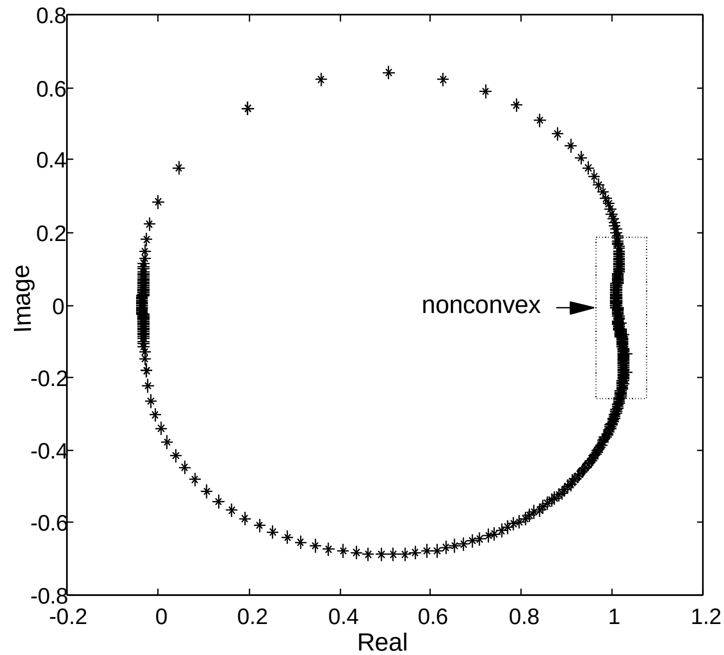


Figure 5: $B_{T_0}$ at 0. 1 rad/sec in the complex plane with $P_0 = 5+0.1i$

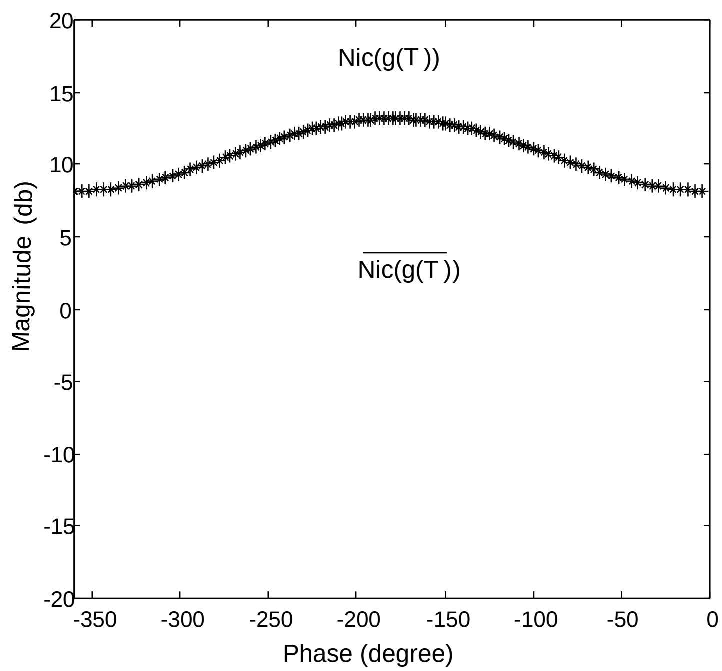


Figure 6: $\overline{n(g(T))}$ for the specification $|S|$ £ $0.285$

For example, a sensitivity reduction specification requiring high-gain is

$$|S| \leq \alpha < 1 \tag{10}$$

(recall $S + T = 1$). We have following result about the convexity of the corresponding QFT bound $B_{T_0}$.

**Corollary 2**: *For the sensitivity reduction specification (10), there always exists a nominal plant* $P_0$ *such that* $B_{T_0}$ *is convex.*

**Proof**: The sensitivity reduction specification (10) is equivalent to

$$|1 - T| \leq \alpha < 1$$

Since $\mathcal{T}$ is a disk centered at 1 with radius less than 1, $n(g(\mathcal{T}))$ is a half plane as shown in Fig 6. It then follows that $\overline{n(g(\mathcal{T}))}$ is also a half plane. Consequently, for any plant template $n(\mathcal{P})$, condition (9) is automatically satisfied. From Corollary 1, we know that there exists a nominal plant $P_0$ such that $B_{T_0}$ is convex. •

In the next section we present a new linear programming algorithm for automatic loop-shaping of QFT controllers. Following results from this section, the algorithm work with closed-loop QFT bounds. It does not yet employ Theorem 1 (work in progress). Its guarantee of internal stability and reduced need for crude approximation of (open-loop) non-convex QFT bounds, constitute its advantage over present automatic loop-shaping algorithms.

## AUTOMATIC LOOP-SHAPING

In this section we formulate automatic loop-shaping of QFT controllers as a linear programming problem.

**Linear Programming.** Given a set of parameters $x = \{x_1, x_2, \ldots, x_n\}$, a linear objective function $f$ and a linear constraint function $g$, the problem of finding the optimal parameters $x$ is stated as

$$\begin{aligned} &\underset{x \in R^n}{\text{minimize}} \quad f(\mathrm{x}) \\ &\text{subject to} \quad g(\mathrm{x}) \le 0 \end{aligned}$$

Linear programming problems can be solved numerically with great efficiency and there exists a large library of software for this purpose (e.g., Branch and Grace, 1996). In a more general setting for control, Boyd and Vandenberghe, (1995) used convex optimization. Recently, Linear Matrix Inequalities (LMIs) have been gaining interest in the control community since many control problems can be formulated as LMIs, and LMIs can be solved exactly by efficient convex optimization algorithms. (e.g., Gahinet et al., 1995).

**Our Problem setup**. The first issue to be addressed is the convexity of the bounds. The results present in the previous section can be used to convexify the bounds (if possible). In such cases, the choice of the nominal plant may not be arbitrary. In cases where the bound $B_{T_0}$ cannot be made convex, we could weaken the specification and use a convex set to approximate it. Alternatively we could strengthen the specification by replacing the bound with the maximum volume ellipsoid contained within it using available software (Veres, 1996). The decision can be made online within QFT's loop-shaping environment. This ellipsoid is then approximated by a set of linear inequalities.

Let the transfer function of T(s) to be synthesized be described by

$$\mathrm{T(s)} = \sum_{j=1}^{n} \frac{\alpha_j}{(s + p_j)} + \sum_{k=1}^{m} \frac{\beta_k s + \gamma_k}{(s^2 + c_k s + d_k)} \qquad (11)$$

Note that while T(s) is linear in its residues ( $\alpha,\beta,\gamma$), it is not linear in its denominator coefficients (or poles). Hence, when (11) is solved using linear programming, the denominator in (11) must be defined a priori. And even though the set of feasible denominators may be large, its selection for linear programming is almost random. This appears to be the single most overriding limitation of our automatic loop-shaping procedure. An expert system for selecting this denominator which is based on the authors' past experiences with manual loop-shaping has been incorporated into the algorithms and shown great promise.

**Relative Degree.** In QFT, it is customary to design only strictly proper controllers (following Horowitz's teachings). In this context, and without loss of generality, we focus on the class of controllers whose relative degree d(C) is at least one (the difference between highest polynomial orders of its denominator and numerator). To insure $d(C) \geq 1$, d(T) must be constrained. Given that $d(P_0) = r$, the requirement on d(T) is (e.g., Section 8.1.3., Helton and Merino, 1994) requires a modification of the above T(s) to

$$T(s) = \prod_{i=1}^{r} \frac{a_i}{(s+a_i)} \left( \sum_{j=1}^{n} \frac{\alpha_j}{(s+p_j)} + \sum_{k=1}^{m} \frac{\beta_k s + \gamma_k}{(s^2 + c_k s + d_k)} \right) \tag{12}$$

where the positive $a_i$ are defined a priori.

**Internal Stability.** To guarantee internal stability, we must guarantee not only stability of $T_0$ but also that the underlying controller C does not share any RHP poles or zeros with the plant family $\mathcal{P}$. One way to achieve this is to add interpolation constraints on T(s) at the RHP poles and zeros of the $P_0$. These constraints are given in the following result (Theorem 11, Chap 8, Helton and Merino, 1994).

*Theorem.* Let $r_l$ (l = 1,…,s) denote the poles and $z_l$ (l = 1,…,t) denote the zeros of the plant P in the closed RHP, so that these poles and zeros have multiplicity $n_l$ (l = 1,…,s) and $m_l$ (l = 1,…,t) respectively. If T is internally stable and d(T) > d(P), then T must satisfy the following interpolation conditions

$$\begin{cases} T(\rho_l) = 1,\ T^{(1)}(\rho_l) = 0, \ldots,\ T^{(n_l-1)}(\rho_l) = 0 \quad (l = 1,\ldots,s) \\ T(z_l) = 0,\ T^{(1)}(z_l) = 0, \ldots,\ T^{(m_l-1)}(z_l) = 0 \quad (l = 1,\ldots,t) \end{cases} \tag{13}$$

**QFT's Optimality Criterion.** Horowitz (1973) defines the optimal QFT controller as the stabilizing controller satisfying its bounds and has minimum high-frequency gain $k_C$ where

$$C(s) \xrightarrow[s\to\infty]{} \frac{k_C}{s^q}, \quad d(C) = q$$

is an *optimal QFT controller.* Define from (12)

$$T(s) \xrightarrow[s\to\infty]{} \frac{k_T}{s^v}, \quad d(T) = v \tag{14}$$

while from (4)

$$T_0(s) \xrightarrow[s\to\infty]{} \frac{k_C}{s^q}\frac{k_{P_0}}{s^e}, \quad d(P_0) = e$$

The controller C is recovered from T(s) in (12) via

$$C = \frac{1}{P_0}\frac{T}{1 - T} \tag{15}$$

then, since $k_{P_0}$ is fixed

$$\min k_T \to \min k_C$$

And since $k_T$ is linear in ( $\alpha,\beta,\gamma$), the QFT's criterion of optimality can be elegantly incorporated in a linear programming formulation.

**Controller Order.** The order of C obtained using (13) is equal to the sum of the order of $P_0$ and the degrees of freedom in the linear programming formulation n+m in (12). This well-known relation is understood to be the price that must be paid for insuring internal stability. Naturally, stable pole/zero cancellations may occur in (13) reducing that order. Here order denotes inter power of the highest-order coefficient in the denominator polynomial.

**Summary.** Before proceeding with numerical examples, let us review the computational steps in our linear program:

1. Convert the open-loop bounds into closed-loop bounds.
2. Check for convexity. If not, either:
   - Compute convex hull, or
   - Compute maximum volume inner ellipsoid.
3. Compute a set of linear inequalities for each bound.
4. Define the poles of T(s) given user defined order.
5. Compute the matrices A and B for the linear inequalities $Ax < B$ from steps 3-4 and (12).
6. Apply internal stability constraint (13) to append rows to A and b.
7. Using the high-frequency gain minimization constraint (14) append a row to A and b. For numerical simplification, we approximate the high-frequency gain by that of T at a very high frequency (where its magnitude is monotonic with respect to frequency).
8. Solve the linear program. If there is no solution allow user to either
   - Select a new set of poles, or
   - Increase order of T.
9. If the problem is solved, extract the controller from (15).

In our program, we have yet to implement the result of Theorem 1 in order to arrive at convex bounds via proper choice of the nominal plant. As it turns out, when the problematic bounds were converted into closed-loop bounds, they were "near" convex. And so, in step 2 we simply computed their convex hull of $B_{T_0}$. Programming Theorem 1 is left for future work.

## AN EXAMPLE

Consider Example 2 from Toolbox Borghesani *et al.*, (1995), with a unity feedback control system and a parametric uncertain plant model described by

$$\mathcal{P} = \left\{ P(s) = \frac{ka}{s(s+a)} : \quad k \in [1,10],\ a \in [1,10] \right\}$$

The problem involves design of a controller C and a pre-filter F to achieve robust stability, a margin specification

$$\left| \frac{P(j\omega)C(j\omega)}{1+P(j\omega)C(j\omega)} \right| \le 1.2, \quad \text{for all } P \in \mathcal{P},\ \omega \ge 0$$

and a tracking specification

$$T_L(\omega) \le \left| F(j\omega) \frac{P(j\omega)C(j\omega)}{1+P(j\omega)C(j\omega)} \right| \le T_U(\omega), \text{ for all } P \in \mathcal{P}, \quad \omega \in [0,10]$$

where

$$T_L(\omega) = \left| \frac{0.6854(j\omega+30)}{(j\omega)^2 + 4(j\omega) + 19.752} \right|, T_U(\omega) = \left| \frac{120}{(j\omega)^3 + 17(j\omega)^2 + 828(j\omega) + 120} \right|$$

Here, we are concerned only with the design of the controller C. The QFT bounds at $\omega =$ [0.01,0.1,0.5,1,2,100] are shown in Figure 7. Also shown are the nominal loop $L_0(j\omega)$ with unity controller ($L_0(j\omega) = P_0(j\omega)$) and the automatically synthesized controller in solid and dashed lines, respectively (the circles ‘o’ denote the response at the above frequencies).

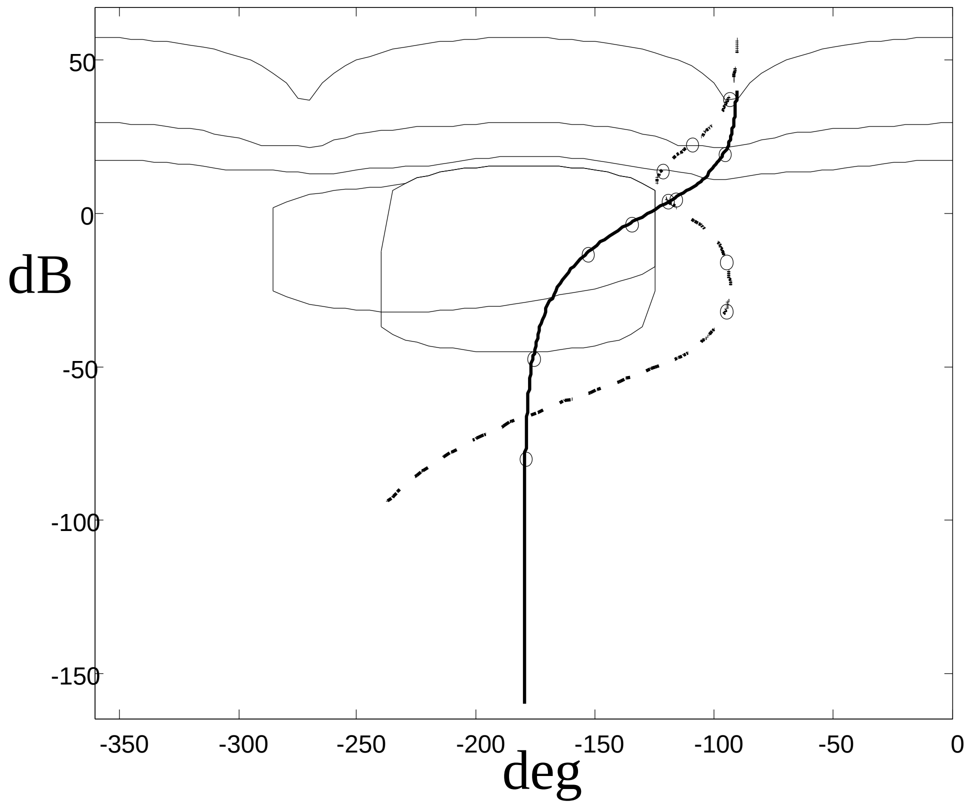


Figure 7: Uncompensated and “optimally” compensated nominal loops with QFT bounds

The optimality of the design is demonstrated in that the loop lies right on the three low-frequency, high-gain type bounds (optimality here is with respect to the linear program setup). It is conceivable that with a different choice of poles of T(s) in (12), an even lower high-frequency gain can be achieved. All computation were carried using in the QFT Control Design MATLAB Toolbox (Borghesani *et al*., 1995). A special GUI was developed for automatic loop-shaping within the Toolbox’s loop-shaping function *lpshape*. This GUI allows for rapid iteration over the values of the poles of T(s) and the desired order of C(s). The design in Figure 6 employed three degrees of freedom (i.e., n+m = 3). Using more degrees of freedom, a lower high-frequency gain in C may be realized. For example, Figure 8 compares the above design (n+m = 3) and a more complex design (n+m = 7), shown in a solid line and dashed line, respectively.

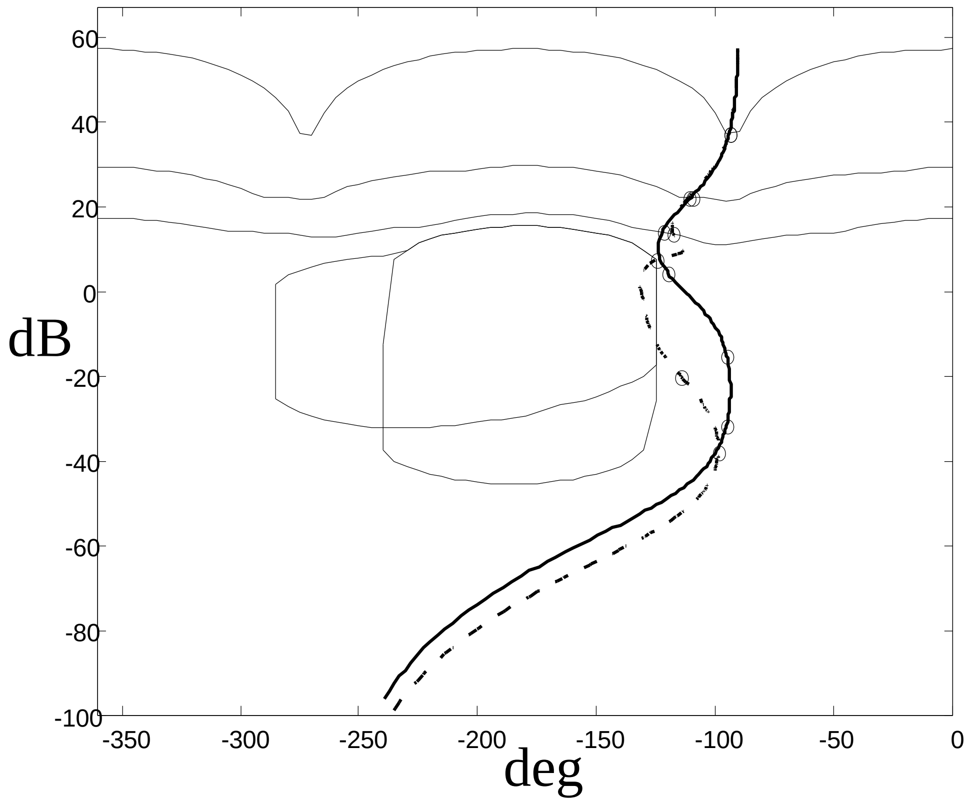


Figure 8: Two “optimally” compensated nominal loops of varying degrees

It is interesting to note that while the second design has a lower high-frequency gain, the loop response violates the margin bounds at a frequency range where bounds are not defined. The optimization scheme cannot guarantee any level of performance at frequencies not included in the formulation. In practice, this does not appear to be a problem since one can add constrains (i.e., bounds) at any number of frequencies. In addition, in this example we did not attempt to exploit the Theorems and used the conservative step of convexifying the two margin-type bounds. However, it should be obvious that given any of the two designs in Figure 8, the designer can focus solely on fine tune in the crossover range. The potential difficulty of obtaining an initial design is removed and the associated cost of conservatism can be greatly reduced using tuning.

**Discussion.** Our linear programming based procedure has been applied successfully to a number of design problems. While our Theorems constitute an important contribution to the understanding of the convexity of QFT bounds, crucial for any linear program, the single limiting factor of our methodology is the need to a priori fix the poles of T(s). It is important to note that the use of automatic loop-shaping in QFT should not be considered the end game. Such a step should be viewed as the initial design to be used by the designer for computer-aided gain-phase tuning, a procedure exploiting the full power of QFT.

## ACKNOWLEDGEMENTS

A number of the ideas in this paper arose from conversations the author was fortunate to have with Carsten Scherer while on a sabbatical leave at TUD, The Netherlands.

## CONCLUSIONS

We have presented a new technique for automatic loop-shaping in QFT. It is based on a linear programming formulation, and by translating the open-loop bounds into certain closed-loop bounds it avoids some of the limitations inherent in previous techniques. In addition, QFT's optimality criterion can elegantly be included in this formulation. A key advantage of this approach is that it provides a definite answer whether a solution exists once the poles and order of T(s) are fixed, and such a solution can be found using efficient numerical algorithms. A sufficient condition for the convexity of the closed-loop QFT bounds is given. While our new procedure is applicable to a large class of problems, further work is required to remove the need to a priori fix the poles of T(s).

## REFERENCES

Bailey, F.N., Helton, J.W, and Merino, O., "Alternative Approaches in Frequency Domain Design of Single Loop Feedback System with Plant Uncertainty, ", *Procs. ACC*, pp. 345-349, 1994.

Ballance, D.J., and Gawthrop, P.J., "Control Systems Design Via a QFT Approach ," *Procs. Control 91*, pp. 476-481, 1991.

Barratt, C., and S. Boyd, S., "Closed-Loop Convex Formulation of Classical and Singular Value Loop Shaping," in *Advances in Control Systems*, C. T. Leondes ed., Vol. 55, 1993.

Borghesani, C., Chait, Y., and Yaniv O., *The QFT Control Design MATLAB Toolbox*, The MathWorks, Inc., Natick, MA, 1995.

Boyd, S., and Vandenberghe, L., *Introduction to Convex Optimization with Engineering Applications*, Lecture notes for EE392X, Spring quarter 1995, Stanford University.

Branch, M.A., and Grace, A., *Optimization Toolbox for MATLAB*, The MathWorks, Natick, MA, 1996.

Bryant, G.F., and Halikias, G.D., "Optimal Loop-Shaping for Systems with Large Parameter Uncertainty via Linear Programming," *Int. J. Control*, Vol. 62, pp. 557-568, 1995.

Gahinet, P., Nemirovski, A., Laub, A.J., and Chilali, M., *LMI Control Toolbox for use with MATLAB*, The MathWorks Inc., Natick, MA, 1995.

Gera, A., and Horowitz, I., "Optimization of the Loop Function," International J. Control, Vol. 31, pp. 389-398, 1980.

Helton, J.W., and Merino, O., *Classical Control Using $H^\infty$ Methods* (rough draft), Dept. of Math, UCSD, 1994.

Horowitz, I., "Optimum Loop Transfer Function in Single-Loop Minimum-Phase Feedback Systems," *International J. Control*, Vol. 18, pp. 97-113, 1973.

Rodrigues, J.M., Chait, Y., and Hollot, C.V., "An efficient algorithm for computing QFT bounds," *J. Dynamic Systems, Measurements, and Control*, Vol. 119, pp. 548-552, 1997.

Thompson, D.F., and O.D.I. Nwokah, "Analytic Loop Shaping Methods in Quantitative Feedback Theory, " *J. Dynamic Systems, Measurement, and Control*, Vol. 116, pp. 169-177, 1994.

Veres, S.M., Ed., *The Geometric Bounding Toolbox for MATLAB*, V5.2, The University of Birmingham, UK, 1996.

Zhao, Y., and Jayasuriya, S., "Robust Stabilization of Uncertain Systems with Parametric Uncertainties," *Procs.12th IFAC Conf.*, Sydney, Australia, Vol. 6, pp. 31-34, 1993.